AFRL-IF-RS-TM-2002-1
In-House Technical Memorandum
November 2002

AUTOMATIC SUMMARIZATION WITH SLOTH (SUMMARIZES LENGTHY DOCUMENTS AND OUTPUTS THE HIGHLIGHTS)

David B. Kaplin

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

[illegible]

APPROVED:

SHARON M. WALTER
Laboratory Program Manager

[illegible]

FOR THE DIRECTOR:

JOSEPH CAMERA, Chief
Information & Intelligence Exploitation Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	**2. REPORT DATE** November 2002	**3. REPORT TYPE AND DATES COVERED** **In-House Tech Memo, Jul 02 – Aug 02**

4. TITLE AND SUBTITLE

AUTOMATIC SUMMARIZATION WITH SLOTH (SUMMARIZES LENGTHY DOCUMENTS AND OUTPUTS THE HIGHLIGHTS)

5. FUNDING NUMBERS

PE - 62702F
PR - 459E
TA - PR
WU - OJ

6. AUTHOR(S)

David B. Kaplin

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

AFRL/IFEA
32 Brooks Road
Rome, NY 13441-4114

8. PERFORMING ORGANIZATION REPORT NUMBER

AFRL-IF-RS-TM-2002-1

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/IFEA
32 Brooks Road
Rome, NY 13441-4114

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

AFRL-IF-RS-TM-2002-1

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Sharon M. Walter/IFEA/315-330-7890. This document reports on a research project pursued by a college student during his summer employment at AFRL/Rome Research Site.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*

SLOTH is an object-oriented, modular, text summarization tool written in the JAVA language. It uses the concept relationship information provided from textual analysis by the eQuery software developed at Syracuse University to create a summary of a text file using extracted sentences. SLOTH stands for Summarizes Lengthy documents and Outputs The Highlights.

14. SUBJECT TERMS war simulation, computer games

Summarization, SLOTH, eQuery, extraction

15. NUMBER OF PAGES 19

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

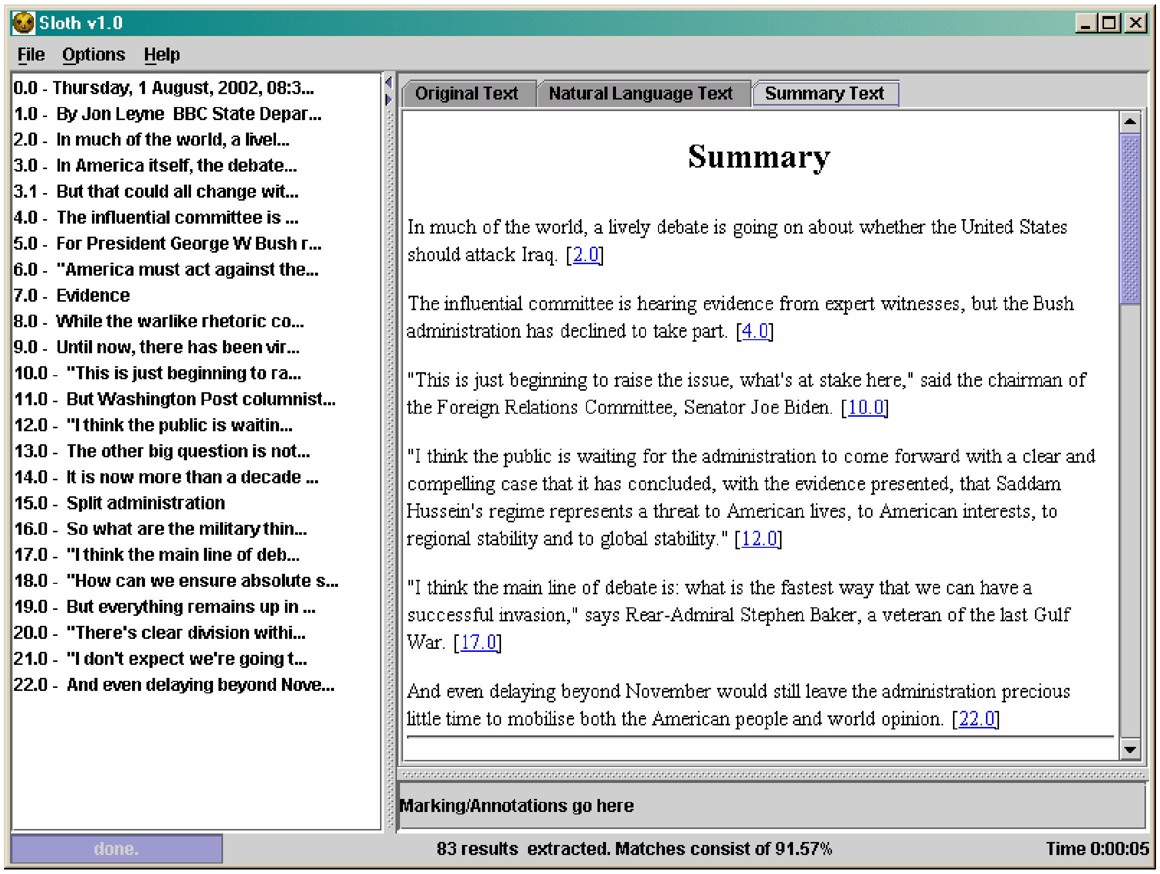
Table of Contents

I. Introduction

SLOTH is an object-oriented, modular, text summarization tool written in the Java software language (version 1.4.0). To aid SLOTH, the program eQuery, developed at Syracuse University, is used to extract important concepts and relations from texts.

SLOTH is capable of producing two forms of summaries, displaying them in either Hypertext (HTML) or Plain Text format. SLOTH operates efficiently on small documents and can handle documents of approximately 80 pages long and larger (tested up to 106 pages).

The SLOTH interface is shown below in Figure 1.



0.0 - Thursday, 1 August, 2002, 08:3...
1.0 - By Jon Leyne BBC State Depar...
2.0 - In much of the world, a livel...
3.0 - In America itself, the debate...
3.1 - But that could all change wit...
4.0 - The influential committee is ...
5.0 - For President George W Bush r...
6.0 - "America must act against the...
7.0 - Evidence
8.0 - While the warlike rhetoric co...
9.0 - Until now, there has been vir...
10.0 - "This is just beginning to ra...
11.0 - But Washington Post columnist...
12.0 - "I think the public is waitin...
13.0 - The other big question is not...
14.0 - It is now more than a decade ...
15.0 - Split administration
16.0 - So what are the military thin...
17.0 - "I think the main line of deb...
18.0 - "How can we ensure absolute s...
19.0 - But everything remains up in ...
20.0 - "There's clear division withi...
21.0 - "I don't expect we're going t...
22.0 - And even delaying beyond Nove...

Summary

In much of the world, a lively debate is going on about whether the United States should attack Iraq. [2.0]

The influential committee is hearing evidence from expert witnesses, but the Bush administration has declined to take part. [4.0]

"This is just beginning to raise the issue, what's at stake here," said the chairman of the Foreign Relations Committee, Senator Joe Biden. [10.0]

"I think the public is waiting for the administration to come forward with a clear and compelling case that it has concluded, with the evidence presented, that Saddam Hussein's regime represents a threat to American lives, to American interests, to regional stability and to global stability." [12.0]

"I think the main line of debate is: what is the fastest way that we can have a successful invasion," says Rear-Admiral Stephen Baker, a veteran of the last Gulf War. [17.0]

And even delaying beyond November would still leave the administration precious little time to mobilise both the American people and world opinion. [22.0]

Marking/Annotations go here

done. 83 results extracted. Matches consist of 91.57% Time 0:00:05

Figure 1: SLOTH Interface

II. SLOTH User Documentation

What is SLOTH?

SLOTH is a text summarization software program. When provided with a text file SLOTH will be able to create a summary of that text using selected sentences from it. SLOTH is platform independent. The tool that SLOTH uses to analyze concepts and relationships between words is called eQuery. eQuery was developed at the Syracuse University Center for Natural Language Processing (website: cnlp.org/tech/equery.asp).

SLOTH stands for

- **S**ummarizes
- **L**engthy documents and
- **O**utputs
- **T**he
- **H**ighlights

Sloth allows easy browsing of the original text of the document, displays the relationships found by eQuery, and outputs the summarization.

How do I use SLOTH?

When SLOTH starts you can

- Read in a text file (File Menu -> Read)
    - This is the file you want to summarize. After selecting the file from the Open File Dialog, SLOTH will begin analyzing your data. While SLOTH is analyzing, you can read the original text of the document. The time needed to analyze a document depends on a few factors:
        - Larger Documents take longer. One 50 page document took approximately 25 minutes to analyze
        - SLOTH **must** initialize the Natural Language Extractor. On the testing machine (1 Ghz Pentium III with 256 MB RAM) this took approximately one minute. Note: T**his process only needs to happen one time while SLOTH is running.**
- Read in a SLOTH Data File
    - Once you have summarized a file with SLOTH you can read in the *SLOTH Data File.* These files have ".sdf" as their ending.

  - On the testing machine it took 7 seconds for a 50 page document to load completely.

Now that a file has been loaded, you can do a lot more!

- The Original Text Tab along with the Index will allow you to browse the document sentence-by-sentence or section-by-section, highlighting your current position in the text.
- Save the summary to the hard drive.
- Change the size of your summary using the preferences dialog box.

What do those numbers mean?

[*Section Number*. *Sentence Number*]
Example:

1002.4 is the location of the Fourth Sentence in the One Thousand and Second Section. For ease of use please use the Document Index List.

What kinds of Documents can I Summarize?

Right now, only Plain Text Document can be Summarized. The Natural Language Parser is limited to the ISO-8859-1 Character Set. All normal ASCII text will work. Some international characters will not.

Where is the "Stop Processing Button"?

Sorry, there is no way to stop analyzing text once the process has started. This is due to Java's thread handling techniques.

How does the Summarization Process work?

Most of the necessary processing takes place while SLOTH is analyzing the structure and grammar of the document. The result of that processing can be stored in a SLOTH Data File and used later to develop a summarization.

Once SLOTH has analyzed the structure and grammar of the document, you can chose one of two different methods for Summarization. These can be found under Options -> Options [Summary Options].

- Sentence Based - The sentences from which eQuery has produced more than a particular number of concepts and relationships will be used to construct the Summary. These sentences will be ordered in the same way as they were in the Original Document.

- Section Based - The entire section is weighed by references from eQuery. They are also listed chronologically. Usually summaries from this method are easier to read than the sentence based variety.

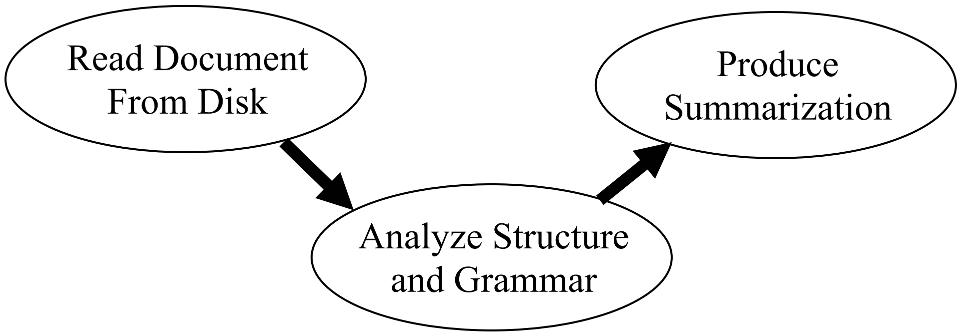


Figure 2: Slow Route to Summarization - Process from Text Files

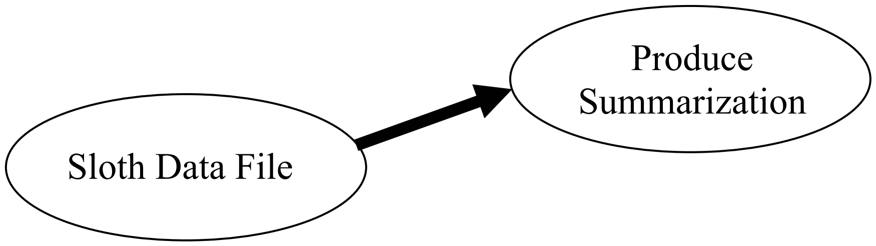


Figure 3: Fast Route to Summarization - Begin with Processed Data

How can I save a Summary?

This option is under the File Menu

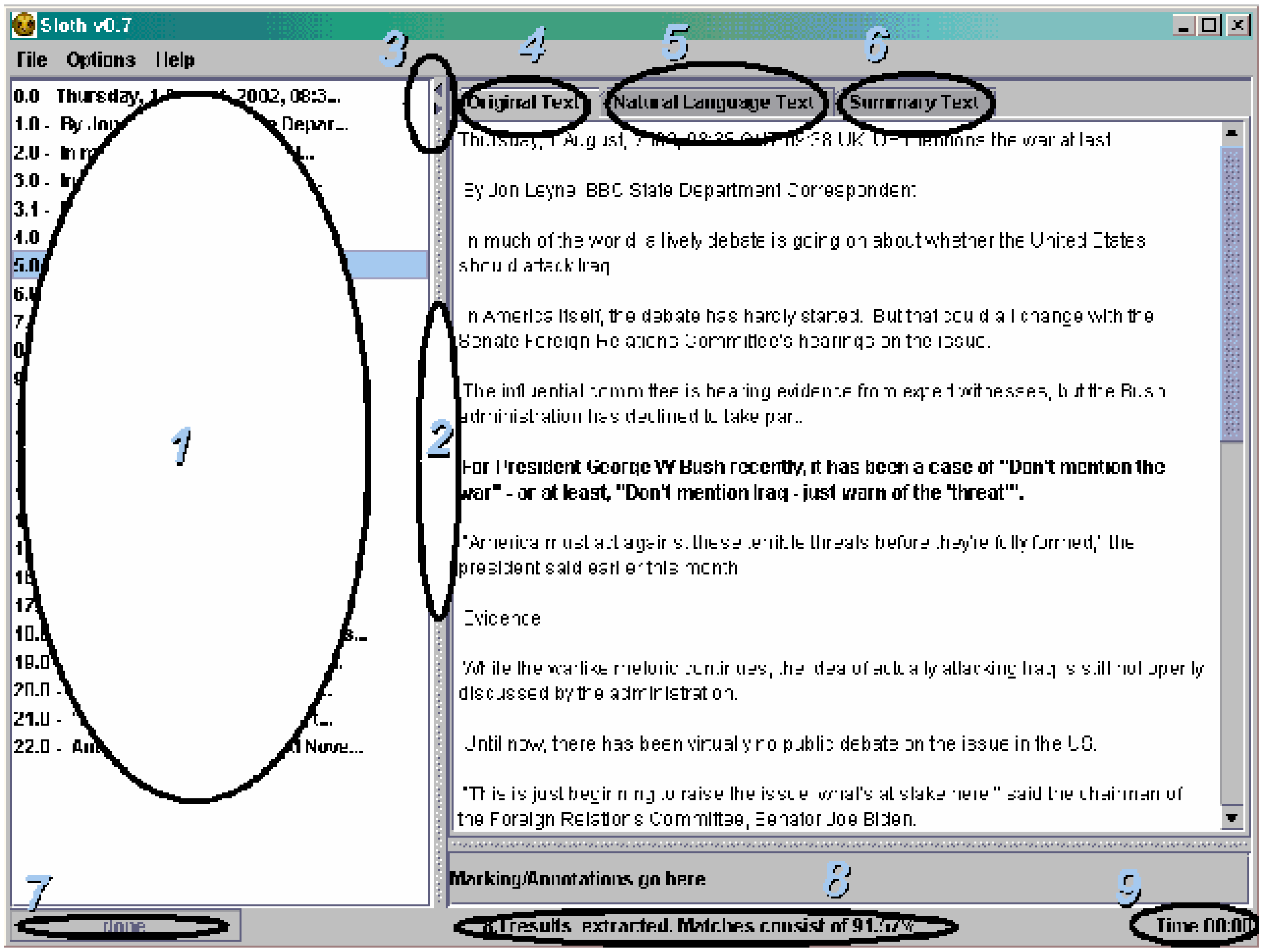
My Document Disappeared, Where did it go?

There is a known problem with the display of large documents within SLOTH. The easiest way to correct this problem is to make sure the document has plenty of line breaks inside of it. If you can, open up the document in a word processor and make sure to add in the line breaks when you save as text. **REMEMBER: When you want to parse this file you MUST make sure the option to Collapse Whitespace is OFF. Your document will not display correctly if this option is turned on.**

Visual Guide to SLOTH

Main Window

The Main Window display is shown in Figure 4, with text identifying the numbered components of the window in the paragraphs below.


Sloth v0.7
File Options Help
3
4
5
6
Original Text
Natural Language Text
Summary Text
0.0 Thursday, 1 [illegible] 2002, 08:3...
1.0 - By Jon [illegible] Depar...
2.0 - In m[illegible]
3.0 - In[illegible]
3.1 -
4.0
5.0
6.0
7.
1
2
7
8
9
done
Marking/Annotations go here
[illegible] results extracted. Matches consist of 91.[illegible]%
Time 00:00

Thursday, 1 August, 2002, [illegible] UK [illegible] mentions the war at last

By Jon Leyne BBC State Department Correspondent

In much of the world a lively debate is going on about whether the United States should attack Iraq

In America itself, the debate has hardly started. But that could all change with the Senate Foreign Relations Committee's hearings on the issue.

The influential committee is hearing evidence from expert witnesses, but the Bush administration has declined to take part.

For President George W Bush recently, it has been a case of "Don't mention the war" - or at least, "Don't mention Iraq - just warn of the "threat"".

"America must act against these terrible threats before they're fully formed," the president said earlier this month

Evidence

While the warlike rhetoric continues, the idea of actually attacking Iraq is still not openly discussed by the administration.

Until now, there has been virtually no public debate on the issue in the US.

"This is just beginning to raise the issue what's at stake here," said the chairman of the Foreign Relations Committee, Senator Joe Biden.

Figure 4: The Main SLOTH Window

1. Document Index

On the left hand side of each entry is its position in the document. The first number is the Section Number and the second is the sentence number in that section. As of this version the definitions of Section and Sentence are used very loosely. If you select one of these entries, it will bold the corresponding text in the document and move to that position within the text.

2. Index Slider

    Dragging this allows you to change the size of the Tabs and the Document Index.

3. Index Slider Move Buttons

    These buttons let you switch your view to the Document Index alone, the Tabs alone, or both at once.

4. Original Text Tab

    Shows the original text of the document. If you see large amounts of whitespace, you should set the option to collapse whitespace.

5. Natural Language Text Tab

    Shows the relationships within the extracted relationships.

6. Summary Text Tab

    Always show the HTML Extended Summary. The original document is included with the summary in front. Selected lines are bolded and referenced in hypertext.

7. Operation Progress

    Briefly explains the status of the operation and its progress. All operations that have an unknown ending time will have a small animation and the elapsed time will be shown.

8. Status Message

    Provides further explanation of the current activity

9. Elapsed Time

    Shows the time that has progressed since the operation began.

File Menu

The File Menu display is shown in Figure 5. Options from the File Menu are listed below.

1. Read Document

    Lets you choose which text file you want to analyze.

2. Save Summary

    - Brief Text Summary - Saves only the Summary (with original positions) to a text file.
    - HTML Summary - Appends the original text to the summary, links them together and bolds statements used in the summary.

3. Load Extracted

    Loads a SLOTH Data Format File for faster summarizing.

4. Save Extracted

    Saves the recently read document as SLOTH Data Format. It is strongly recommended to save your results to avoid analyzing long documents twice.

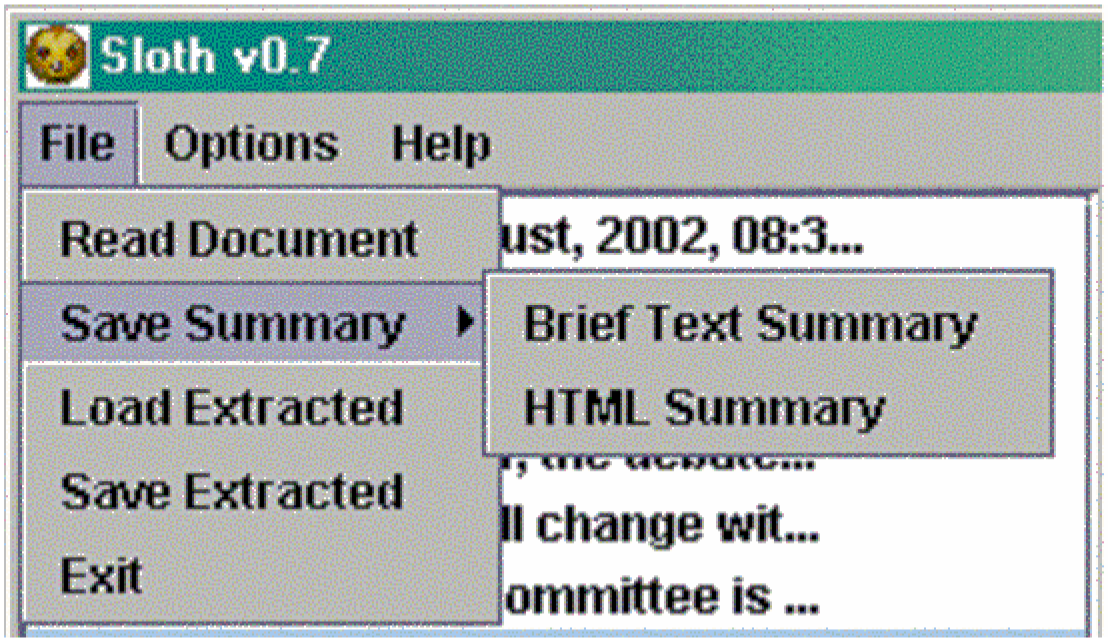


Figure 5: The File Menu for SLOTH

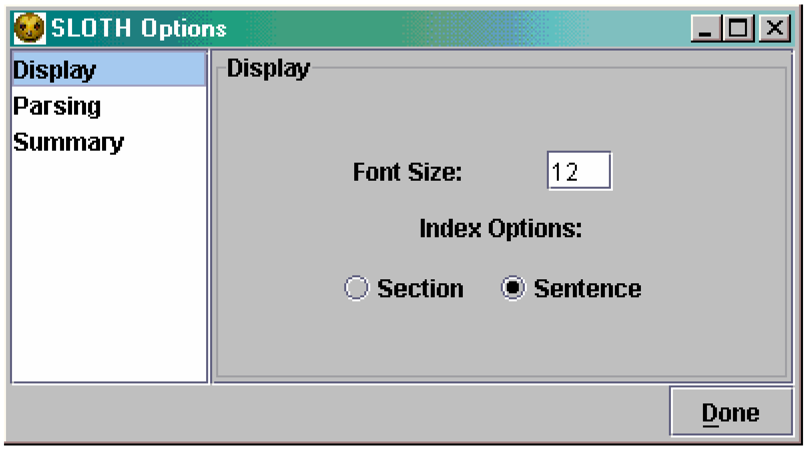
Options

Instructions to SLOTH for Display, Parsing, and Summary options are set from this window.

Display

The Font Size affects the Document Index and the Original Text panels. The Font Size cannot be applied to the Natural Language Text or the Summarized Text.

Index Options allow you to configure the Document Index to show only the first sentence in each Section or in every Sentence.

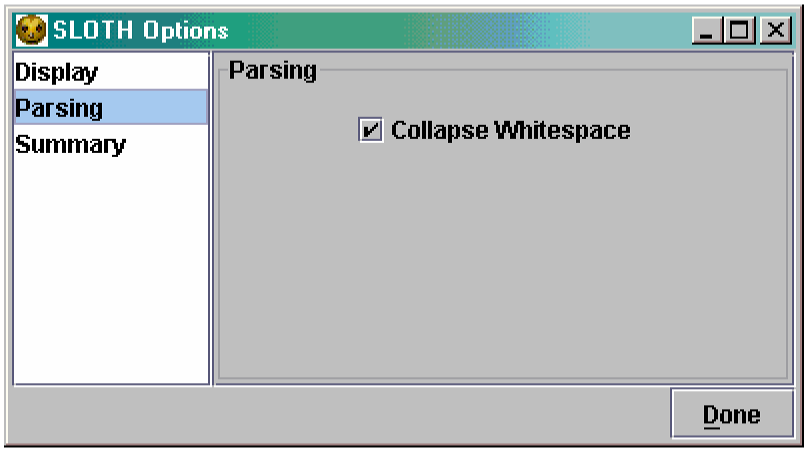


Font Size: 12

Index Options: Section [✓ Sentence]

Figure 6: SLOTH Options for the Display

Parsing

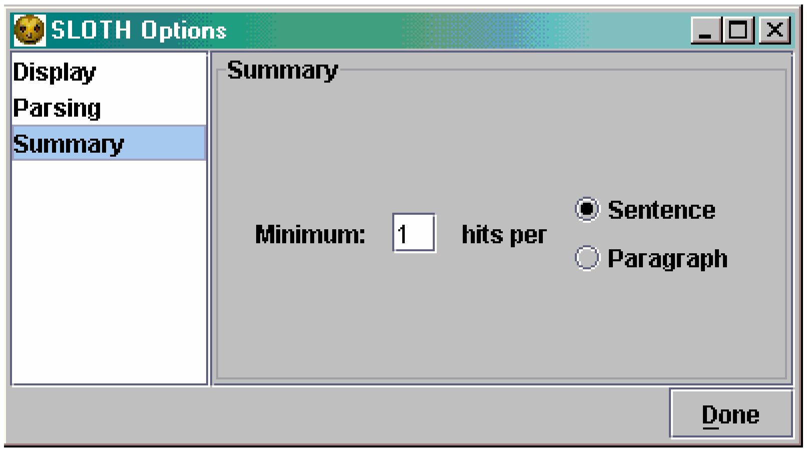


✓ Collapse Whitespace

Figure 7: SLOTH Option for Parsing

When enabled, the Collapsing Whitespace option, makes the document look a lot cleaner when displayed in the Original Text Tab. If this option is enabled then "John Smith" and "John   Smith" will both become "John Smith".

Summary


SLOTH Options

Summary

Minimum: 1 hits per [✓ Sentence] Paragraph

Figure 8: SLOTH Options for Summary

Each **hit** corresponds to the number of relationships found per indexed part of a document. The number of hits usually corresponds with the number of definitions, or strong language used in that part of the document. These are sections that *could* be important to the summary. By setting a **minimum number** of hits a summary can be thinned down or increased.

If the **Paragraph** choice is selected then all of the sections will be weighed by their total number of hits. These summaries tend to flow better because each Section is extracted in its entirety from the document.

III. Technical Information

Internal States

To keep track of program flow, SLOTH has a set of internal states that dictate what options are valid and that manipulate the interface for consistency. The operation of SLOTH follows these paths strictly. If there is an error found within the operation of SLOTH it can be traced to one link within this System.

Figure 9 depicts the SLOTH program flow.

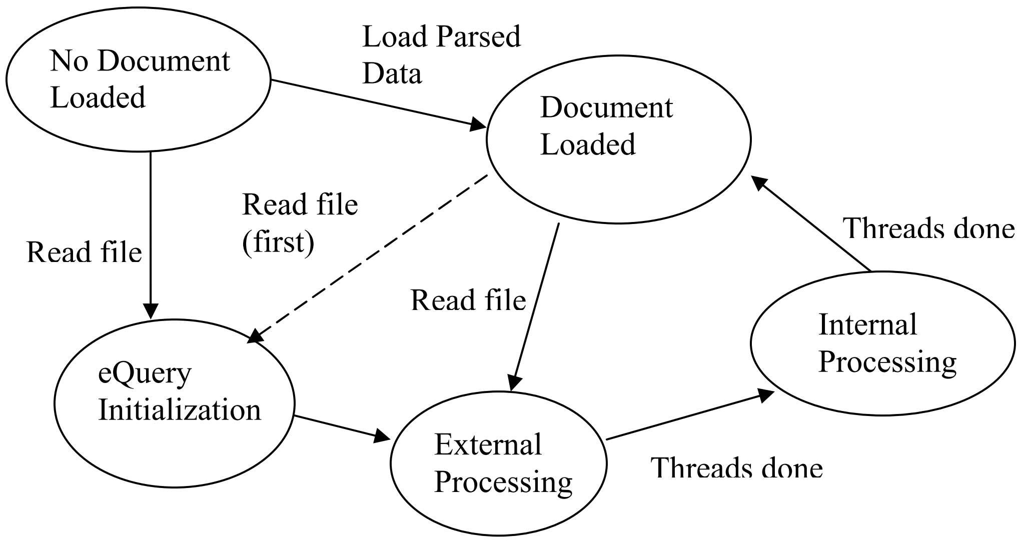


Figure 9: SLOTH Internal States

SLOTH Packages

Model

The classes here are as close to classical data structures as possible.
Virtually all algorithms are implemented using classes within the controller package. The NLPStat class does a little more than normal storage by also encapsulating a couple of algorithms for finding particular relationships among the data

View

This package is about the interpretation of data. Through abstraction, these classes make file I/O easy. To get data into the right form for the Natural Language Processor the DocumentBackbone must be transformed into an XML-like format. In particular the character set must be ISO-8859-1 (currently a common choice for most web-browsers). Using a similar interface, summaries are output to files.
This package also takes care of the writing of Parsed Data to its serialized data form.

Pref

This package provides a common interface for classes to interact with program Preferences.

Pref.Modules

This package contains concrete examples of AbstractPrefModule.

Controller

The classes within controller tie the program together. Several thread classes are responsible for communicating to eQuery along with file input.

SlothManager stores the finite-state automata that represents the program flow.

DocumentComposite handles the more important functions that provide the data for many parts of the program.

The SubjectThread, SubjectListener, and Trigger system allow easy tracking of threads in a system that requires things to be finished.

GUI

This is a system of objects that get manifest in the class GUIComposite. All of the classes interface with the Java Swing Framework to provide a user interface. A lot of work was done to transfer most of the "controller" type operations into separate classes in the package controller. These items in the gui package do not have a particular amount of intelligence to them. They are here primarily for display.

When an interaction does take place the gui component usually talks to its counterpart in the controller package or is told exactly what to do from that controller.

SLOTH Runtime Statistics

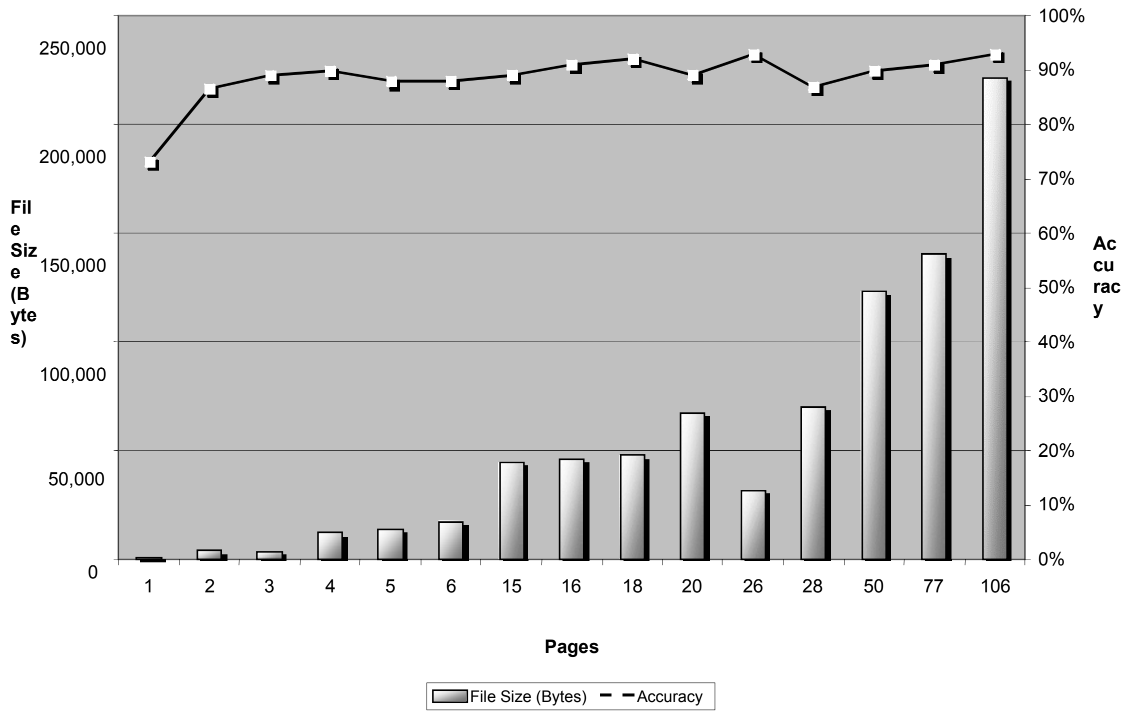


Figure 10: SLOTH Statistics

Pages	Approximate Time
Under Five	30 seconds
Under Fifteen	3 Minutes
Under Thirty	6 Minutes
Under Fifty	30 Minutes
Under Eighty	40 Minutes
Under One Hundred and Ten	1 Hour and 15 Minutes

Figure 11: Average Execution Times on a 1Ghz PIII with 261 MB RAM (raw documents)

IV. Using Syracuse University's eQuery Software

The Natural Language Text is provided for SLOTH from a program called eQuery from Syracuse University's Center for Natural Language Processing. Information about eQuery can be found at http://www.cnlp.org/

When eQuery examines a document it outputs three bits of extracted information:

- Entity - Usually a noun, sometimes a verb
- Relationship - How the entity is related to the information
- Information - More information about the Entity

To better explain the output of the Natural Language Extractor I will show this example:

Original Text:

"This is a test of the lexical capabilities of SLOTH. What sentences will it want to highlight? I really have no idea what the outcome will be. But, SLOTH does! And now the amazing results...

Drumroll Please...

Ta Da!"

Natural Language Text:

#	Entity	Relationship	Information	Ref	Explanation
0	test	object	lexical capability	-1.0	Notice the first few entries have -1.0 as their Reference. This number indicates that SLOTH was not able to find both the entity and the information within the same sentence. In this case, the reason is because the NLE changed the noun agreement. However, this is minimized a little bit of guesswork within SLOTH. Notice Entry #7. the word "result" does not appear, but because "result" can be seen as a form of results SLOTH was able to track down the original sentence of origin.
1	test	associated	lexical capability	-1.0	
2	capability	characteristic	lexical	-1.0	
3	capability	associated	SLOTH	-1.0	
4	want	agent	it	0.1	
5	have	agent	I	0.1	
6	have	object	idea	0.1	
7	result	characteristic	amazing	0.2	

eQuery provides information from the text in the form of a tab-delimited output String:

```
Entity     Relationship     Information
```

Things to remember about this string:

1. Fields do not retain the tense originally found in the document
2. Fields do not retain the noun-plural agreement.

3. Occasionally the entity and information fields appear to be switched.
4. The Entity field may contain one of many custom entities. These are signified with an underscore character. For example, "going on" becomes "go_on"
5. The order in which eQuery returns the result sometimes has to do with the original order of the document. This is not a rule. There have been plenty of occasions in which eQuery returns something from the bottom of the document first. This aspect is the reason why SLOTH needed its document ordering.
6. eQuery will not return any results if all of the text is in the same case.

V. SLOTH and Memory Usage

Natural language extraction and processing are very intense and demanding operations. SLOTH is no exception to this rule. In its windows implementation, the script file that launches SLOTH alerts the Java Virtual Machine (JVM) that SLOTH demands a maximum of 128MB RAM for its heap. Although this extra allocation will do little to speed up the process of analyzing documents, it does allow SLOTH to handle larger documents that it could if it was using the normal 64MB heap allotted from the Windows JVM.

After parsing a large document SLOTH retains more memory than it needs. In future revisions this may be fixed by calling the garbage collector after a document is finished being parsed. Another way to alleviate this problem on computers with smaller amounts of RAM available is to save the extracted information. The amount of memory allocated to use SLOTH data files is significantly less than using directly parsed data.

SLOTH Files

userprefs.data – holds information, like Summary Method, Font Size …
Structure: This is a Serialized HashMap of all the preferences.
*.sdf - SLOTH [Parsed] Data File
Structure: Serialized String Header, DocumentComposite.

NOTE: Serialization is compatible with future versions of the Java language, however, if there are any changes within the classes:
DocumentBackbone
DocumentComposite
NLPAdvocate
NLPStat
Then all previously created SLOTH Data Files will be incompatible with future versions.

There are two known Exceptions thrown while SLOTH is in memory.
One is because of a flaw in the JProgressBar class. It is benign; no operation is hindered by this exception

The second is a flaw in the line breaking algorithms used in Java 1.4 found in both JEditorPane and JTextPane. Both of Original Text Pane and Summary Pane are affected by this flaw. They will not display data when there are too many characters per line. The actual computational parts of SLOTH are not affected by this flaw, only these two display components. The only fix is to make sure that the large input documents have a decent number of line breaks. **Make sure to turn Collapse Whitespace OFF.** That particular option removes what are viewed to be extraneous line breaks.

VI. How to Extend SLOTH

Since SLOTH was made to be modular there are quite a few areas that can be modified to add to the functionality of SLOTH. Some possible areas of extension are listed here:

- Improving Text Analysis:
  Add an option for "Pureeing" large documents into chunks. Perhaps a "chop" option to cut documents into eighths, or sixteenths, merged into a DocumentBackbone. This would allow a faster analysis and, given current algorithms, not affect the results much.

- Parsing other file types:
  The class in charge of parsing documents into DocumentBackbone form is called PlainParser, inherited from AbstractParser in the package controller. Suggested other file types are HTML, PDF, and Word Document. (The latter may be tricky.)

- Output to other file types:
  AbstractFileView and AbstractStatisticalSummaryView are good classes to extend. Other possible output types: Database, XML, other forms of HTML Summary, based on different parts of the document.

- Adding new preferences
  pref.AbstractPreferenceModule allows an easy interface to implement new preferences. PreferenceChangeListener also plays a big role in changing preferences, using the observer pattern.

Remember that any change in the GUI will most likely involve changing both GUIComposite and SlothManager. More complex additions may need new internal states within SlothManager.